

# Examen de Programmation Réseaux

Juliusz Chroboczek

9 janvier 2018

La durée de l'examen est de 2 heures. Les documents sont autorisés, le matériel électronique est interdit. Le sujet consiste de 3 pages.

**Question 1.** Mon appartement et le laboratoire d'Informatique de l'Université Paris-Diderot sont interconnectés par une liaison point-à-point ayant un débit de 100 Mbit/s et un *round-trip time* (RTT) de 1.2 ms. Un faucon (une espèce d'oiseau) portant une carte SD de 32 Go peut faire l'aller retour entre chez moi et mon bureau en 12 minutes. Je ne possède qu'un seul faucon, et qu'une seule carte SD de 32 Go.

1. Quel est le temps nécessaire pour envoyer un paquet de 1500 octets de données et recevoir un acquittement par la liaison point-à-point ? Même question si on utilise le faucon. (N'oubliez pas de multiplier ou de diviser par 8 s'il le faut.)
2. Quel est le temps nécessaire pour envoyer 1 Go de données ( $10^9$  octets) par la liaison point-à-point et recevoir un acquittement ? (On pourra supposer qu'on peut envoyer les données sans interruption et sans attendre d'acquittements intermédiaires, i.e. que la fenêtre est effectivement infinie, et qu'aucune donnée n'est perdue.) Même question si on utilise le faucon.
3. Mêmes questions pour 1 To de données ( $10^{12}$  octets).
4. Comparez les performances de la liaison point-à-point et du faucon.

**Question 2** (Bit de parité sur Hamming). Dans toute cette partie, on considère le code suivant : étant donné un mot  $w$  de  $m$  bits, on applique d'abord le code de Hamming à  $w$ , ce qui donne le mot  $w'$ , puis on rajoute à la fin de  $w'$  son bit de parité ( $w'' := w'.b$  avec  $b := |w'|_1 \% 2$ ), ce qui donne le mot  $w''$  qui est transmis.

Pour les questions qui suivent, il n'y a pas à prouver les propriétés ni à détailler les algorithmes de codage et décodage du bit de parité ainsi que du code de Hamming, vus en TD. Cependant, il faut les citer explicitement si nécessaire.

1. Codez  $w = 01101011010$  selon le codage défini ci-dessus (i.e. il faut donner le  $w''$  correspondant) en détaillant les étapes.
2. Montrez que ce code détecte 3 erreurs. Donnez un exemple où 4 erreurs arrivent sans être détectées.
3. On admettra que ce code corrige une erreur mais pas deux (vous n'avez pas à le démontrer, vous l'admettez). Donnez le pseudo-code d'un algorithme de décodage qui donne aussi une

estimation du nombre d'erreurs qui ont eu lieu (en supposant qu'il y a eu moins de 4 erreurs) et transmet soit le mot décodé soit « erreur incorrigible ».

4. Quel est le nombre de bits ajoutés par le codage? Comparez avec le codage de Hamming en termes (1) de puissance de détection, (2) de puissance de correction, et (3) de nombre de bits ajoutés.

### Question 3 (Analyse de traces).

1. Dans la trace suivante, indiquez (1) le protocole de couche réseau utilisé, (2) le protocole de couche transport, (3) le protocole de couche application, (4) la taille maximale de segment utilisée (à l'exclusion des entêtes de couche réseau et transport), (5) sur quel pair la trace a été capturée, (6) le RTT (la latence) de la communication, et (7) le débit moyen.

```
19:08:48.780743 IP6 2a01:e0c:1:1598::2.80 > 2001:660:3301:9202::ac17:2436.47722:
  Flags [.], seq 56161:57601, ack 135, win 29480, length 1440
19:08:48.780747 IP6 2a01:e0c:1:1598::2.80 > 2001:660:3301:9202::ac17:2436.47722:
  Flags [.], seq 57601:59041, ack 135, win 29480, length 1440
19:08:48.780750 IP6 2001:660:3301:9202::ac17:2436.47722 > 2a01:e0c:1:1598::2.80:
  Flags [.], ack 59041, win 64800, length 0
19:08:48.780753 IP6 2a01:e0c:1:1598::2.80 > 2001:660:3301:9202::ac17:2436.47722:
  Flags [.], seq 59041:60481, ack 135, win 29480, length 1440
19:08:48.780755 IP6 2a01:e0c:1:1598::2.80 > 2001:660:3301:9202::ac17:2436.47722:
  Flags [.], seq 60481:61921, ack 135, win 29480, length 1440
19:08:48.780759 IP6 2001:660:3301:9202::ac17:2436.47722 > 2a01:e0c:1:1598::2.80:
  Flags [.], ack 61921, win 64800, length 0
19:08:48.780761 IP6 2a01:e0c:1:1598::2.80 > 2001:660:3301:9202::ac17:2436.47722:
  Flags [.], seq 61921:63361, ack 135, win 29480, length 1440
19:08:48.780764 IP6 2a01:e0c:1:1598::2.80 > 2001:660:3301:9202::ac17:2436.47722:
  Flags [.], seq 63361:64801, ack 135, win 29480, length 1440
19:08:48.780768 IP6 2001:660:3301:9202::ac17:2436.47722 > 2a01:e0c:1:1598::2.80:
  Flags [.], ack 64801, win 64800, length 0
19:08:48.780770 IP6 2a01:e0c:1:1598::2.80 > 2001:660:3301:9202::ac17:2436.47722:
  Flags [.], seq 64801:66241, ack 135, win 29480, length 1440
19:08:48.782728 IP6 2a01:e0c:1:1598::2.80 > 2001:660:3301:9202::ac17:2436.47722:
  Flags [.], seq 66241:67681, ack 135, win 29480, length 1440
19:08:48.782735 IP6 2001:660:3301:9202::ac17:2436.47722 > 2a01:e0c:1:1598::2.80:
  Flags [.], ack 67681, win 64800, length 0
```

2. Dans la trace suivante, indiquez (1) quel est le protocole de couche transport utilisé, (2) qui est le client au sens de la couche transport.

```
19:13:42.625627 IP6 2001:660:3301:9202::ac17:2436.33462 > 2001:660:3301:9200::51c2:1b9b.22:
  Flags [SEW], seq 3865287329, win 28800, options [mss 1440,nop,nop,sackOK], length 0
19:13:42.626072 IP6 2001:660:3301:9200::51c2:1b9b.22 > 2001:660:3301:9202::ac17:2436.33462:
  Flags [S.E], seq 697748573, ack 3865287330, win 28800, options [mss 1440,nop,nop,sackOK],
  length 0
19:13:42.626094 IP6 2001:660:3301:9202::ac17:2436.33462 > 2001:660:3301:9200::51c2:1b9b.22:
  Flags [.], ack 1, win 28800, length 0
19:13:42.626453 IP6 2001:660:3301:9202::ac17:2436.33462 > 2001:660:3301:9200::51c2:1b9b.22:
  Flags [P.], seq 1:34, ack 1, win 28800, length 33
19:13:42.626723 IP6 2001:660:3301:9200::51c2:1b9b.22 > 2001:660:3301:9202::ac17:2436.33462:
  Flags [.], ack 34, win 28800, length 0
```

#### Question 4.

1. Pourquoi le protocole UDP est-il plus adapté que TCP à la transmission de flux vidéo ?
2. Le site de vidéos *YouTube* utilise TCP comme couche de transport. Donnez deux raisons qui ont pu pousser les ingénieurs de *YouTube* à choisir TCP plutôt que UDP, et expliquez comment *YouTube* contourne le problème que vous avez décrit à la question précédente.
3. Depuis quelques années, *YouTube* utilise une technologie nommée *DASH* (*Dynamic Adaptive Streaming over HTTP*) où la vidéo est découpée en fragments de quelques secondes, dont chacun est transmis au-dessus de TCP, et le client (le navigateur) fait une requête pour le fragment suivant juste avant la fin du fragment courant. Donnez une ou deux raisons qui ont pu pousser au déploiement de *DASH*.